IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Current Listing of Claims

What is claimed is:

1. (Currently amended) A method comprising:

transferring information between a target device and a first buffer which is one of a plurality of buffers, the first buffer being pointed to by a current buffer value stored in a controller;

adjusting the current buffer value stored in the controller to point to a next buffer if the current buffer value is different than a last buffer value stored in the controller; ~~and~~

servicing one of the plurality of buffers utilizing one of the current buffer value and the last buffer value from the controller, wherein the current buffer value is a first index into a buffer descriptor table and wherein the current buffer value points to the first buffer by pointing to a buffer descriptor offset by the first index from a buffer descriptor table base location; and

prefetching a next buffer descriptor from the buffer descriptor table.

2. (Original) The method of claim 1 wherein servicing comprises:

retrieving the current buffer value from the controller;

comparing the current buffer value to a head pointer of a buffer list;

marking a buffer pointed to by the head pointer as being processed if the head pointer has not reached the current buffer value; and

adjusting the head pointer to point to a next potentially reclaimable buffer if the head pointer has not reached the current buffer value.

3. (Original) The method of claim 1 wherein servicing comprises:

preparing a buffer pointed to by an incremented tail pointer; and

storing the incremented tail pointer as the last buffer value in the controller.

4. Canceled.

5. Canceled.

6. (Original) The method of claim 1 further comprising:

testing an interrupt field to determine whether the interrupt field is set to a first value to interrupt upon completion of processing the first buffer; and

generating an interrupt if the interrupt field is set to the first value.

7. (Original) The method of claim 1 further comprising:

executing a buffer underrun routine according to a command field associated with the first buffer if there are no further buffers available for transfer.

8. (Currently amended) A bus agent comprising:

a current buffer register for storing a first value indicating a first memory location for a current buffer, wherein the current buffer register contains a first index value and wherein the first index value in the current buffer register indicates

the first memory location by pointing to a first buffer descriptor in a buffer descriptor table, the first buffer descriptor being the first index value locations from a buffer descriptor table base;

a last buffer register for storing a second value indicating a second memory location for a last buffer ready for processing, wherein the last buffer register contains a second index value and wherein the second index value in the last buffer register indicates the second memory location by pointing to a second buffer descriptor in the buffer descriptor table, the second buffer descriptor being the second index value locations from the buffer descriptor table base; ~~and~~

control logic coupled to transfer data to or from the current buffer and to update the current buffer register to point to a next buffer unless the first value from the current buffer register is equivalent to the second value from the last buffer register; and

a prefetch buffer register, wherein the control logic is coupled to set the current buffer register equal to a value in the prefetch buffer register, to increment the value in the prefetch buffer register, and to schedule a prefetch of a buffer descriptor pointed to by the prefetch buffer register.

9. Canceled.

10. Canceled.

11. Canceled.

12. (Original) The bus agent of claim 8 wherein the current buffer register contains a first pointer to a linked list of buffers in memory and wherein the last buffer register contains a second pointer to a last buffer in the linked list of buffers in memory.

13. (Original) A system comprising:

a bus controller comprising:

a current buffer register for storing a first value indicating a memory location of a current buffer;

a last buffer register for storing a second value indicating a memory location of a last buffer, the last buffer being the last buffer which is prepared for processing; and

control logic coupled to transfer data to or from the current buffer and to update the current buffer register to point to a next buffer unless the first value is equivalent to the second value;

a processor; and

a memory coupled to the processor, the memory containing:

a plurality of buffers including the current buffer and the last buffer;

at least one software routine which, if executed, causes the system to service at least one of the plurality of buffers utilizing information stored in one of the current buffer register and the last buffer register.

14. (Original) The system of claim 13 wherein the memory further contains:

a buffer descriptor table having a plurality of buffer descriptors pointing to the

plurality of buffers, and

wherein the current buffer register contains a first index value, the first index value in the current buffer register indicating the current buffer by pointing to a first buffer descriptor in the buffer descriptor table, the first buffer descriptor being the first index value locations from a buffer descriptor table base.

15. (Original) The system of claim 14 wherein the last buffer register contains a second index value, the second index value in the last buffer register indicating the last buffer by pointing to a second buffer descriptor in the buffer descriptor table, the second buffer descriptor being the second index value locations from the buffer descriptor table base.

16. (Original) The system of claim 13 wherein the current buffer register contains a first pointer to a linked list of buffers in memory and wherein the last buffer register contains a second pointer to a last buffer in the linked list of buffers in memory.

17. (Original) The system of claim 13 wherein the bus controller further comprises a prefetch buffer register and wherein the control logic is further coupled to set the current buffer register equal to a value stored in the prefetch buffer register, to increment the prefetch buffer register, and to schedule a prefetch of a prefetch buffer descriptor pointed to by the prefetch buffer register.

18. (Original) The system of claim 13 wherein the at least one software routine

comprises:

a preparation routine which, if executed, causes the system to perform:

preparing a buffer pointed to by an incremented tail pointer; and

storing the incremented tail pointer in last buffer register.

19. (Original) The system of claim 13 wherein the at least one software routine comprises:

a reclamation routine which, if executed, causes the system to perform:

retrieving a current buffer value from the current buffer register;

comparing the current buffer value to a head pointer of a buffer list;

marking a buffer pointed to by the head pointer as being processed if the head pointer has not reached the current buffer value; and

adjusting the head pointer to point to a next potentially reclaimable buffer if the head pointer has not reached the current buffer value.

20. (Original) A system comprising:

a processor;

a bus agent comprising:

a current index register;

a prefetch index register; and

a last index register;

a buffer descriptor base address register;

control logic coupled to transfer data to or from a buffer pointed to by the current index register, to stop retrieving data if the current index register

and the last index register contain equivalent values, to set the current index register equal to the prefetch index register, to increment the prefetch index register, and to schedule a prefetch of a prefetch buffer descriptor pointed to by values stored in the prefetch index register and the buffer descriptor base address register.

a memory coupled to the processor and the bus agent containing:

a table storing a plurality of buffer descriptors

a preparation routine which, if executed, causes the system to perform:

checking an incremented tail index to determine whether the incremented tail index points to a free buffer;

if the incremented tail index points to the free buffer, then

preparing the free buffer by storing data to be retrieved by the bus master into the free buffer;

storing the incremented tail index in the last index register;

a reclamation routine which, if executed, causes the system to perform:

retrieving a current index value from the current index register;

comparing the current index value to a head pointer;

if the head pointer is less than the current index value, then

marking a buffer pointed to by the head pointer as free;

incrementing the head pointer;

returning to comparing the current index value to the head pointer.

21. (Currently amended) An article comprising a machine readable medium having stored

thereon a plurality of instructions which, if executed by the machine, cause the machine to perform:

transferring information between a direct memory access (DMA) controller and a first buffer which is one of a plurality of buffers, the first buffer being pointed to by a current buffer register in the DMA controller;

adjusting the current buffer register to point to a next buffer if the current buffer register contains a different value than a last buffer register; ~~and~~

servicing one of the plurality of buffers utilizing information contained in one of the current buffer register and the last buffer register, wherein a value in the current buffer register is a first index into a buffer descriptor table and wherein the value in the current buffer register points to the first buffer by pointing to a buffer descriptor offset by the first index from a buffer descriptor table base location; and

prefetching a next buffer descriptor from the buffer descriptor table.

22. (Original) The article of claim 21 wherein the servicing performed by the machine further comprises:

retrieving a first value from the current buffer register;

comparing the first value to a head pointer of a buffer list;

marking the buffer pointed to by the head pointer as being processed if the head pointer has not reached the first value; and

adjusting the head pointer to point to a next potentially reclaimable buffer if the head pointer has not reached the first value.

23. (Original) The article of claim 21 wherein the servicing performed by the machine further comprises:

preparing a buffer pointed to by an incremented tail pointer; and

storing the incremented tail pointer in last buffer register.